Realtime Mobile Bandwidth and Handoff Predictions in 4G/5G Networks

Lifan Mei[a,*], Jinrui Gou[a], Yujin Cai[a], Houwei Cao[b] and Yong Liu[a]

[a]NYU WIRELESS, Department of Electrical and Computer Engineering, New York University, Brooklyn, NY 11201, USA
[b]Department of Computer Science, New York Institute of Technology, New York, NY 10023, USA

ARTICLE INFO

Keywords:
Bandwidth Prediction
Handoff Prediction
Deep Learning
5G
Measurement
Public Transportation Scenario

ABSTRACT

Mobile apps are increasingly relying on high-throughput and low-latency content delivery, while the available bandwidth on wireless access links is inherently time-varying. The handoffs between base stations and access modes due to user mobility present additional challenges to deliver a high level of user Quality-of-Experience (QoE). The ability to predict the available bandwidth and the upcoming handoffs will give applications valuable leeway to make proactive adjustments to avoid significant QoE degradation. In this paper, we explore the possibility and accuracy of realtime mobile bandwidth and handoff predictions in 4G/LTE and 5G networks. Towards this goal, we collect long consecutive traces with rich bandwidth, channel, and context information from public transportation systems. We develop Recurrent Neural Network models to mine the temporal patterns of bandwidth evolution in fixed-route mobility scenarios. Our models consistently outperform the conventional univariate and multivariate bandwidth prediction models. For 4G & 5G co-existing networks, we propose a new problem of handoff prediction between 4G and 5G, which is important for low-latency applications like self-driving strategy in realistic 5G scenarios. We develop classification and regression based prediction models, which achieve more than 80% accuracy in predicting 4G and 5G handoffs in a recent 5G dataset.

arXiv:2104.12959v1 [cs.NI] 27 Apr 2021

1. Introduction

The growth of mobile Internet traffic has accelerated in the recent years, thanks to both the breakthroughs in wireless access technologies, such as mmWave and massive MIMO, and a fast-growing array of mobile multimedia apps, ranging from video streaming/conferencing, Virtual Reality, Augmented/Mixed Reality, to autonomous driving, etc. While the next-generation mobile access infrastructure, such as 5G network, is designed to deliver high-throughput, low-latency, and high-reliability, the actual Quality-of-Service delivered to users is still vulnerable to various impairments to the physical channel quality between user devices and access points. It is well-known that wireless signals can be attenuated by interference, path loss, static and mobile blockage, etc. The current 4G/LTE mobile access is much more volatile and unpredictable than WiFi and wireline accesses. While 5G mmWave transmission can deliver data rates over $1Gbps$, mmWave signals at higher frequency bands (20 -100 GHz) incur higher free-space path loss, blockage loss, and penetration loss [1]. The bandwidth variations experienced by users of the initial batch of commercial 5G deployments, both mmWave and Sub-6GHz, are much more dramatic than 4G/LTE [20, 22].

How to deliver a high level of user Quality-of-Experience under volatile mobile access conditions is a main challenge for mobile app developers and multimedia application service providers. For delay-tolerant applications, bandwidth variations can be "absorbed" by sacrificing application-level latency. In the example of on-demand video streaming, a video buffer of tens of seconds is typically employed so that video can be streamed smoothly as long as the video rate matches the average bandwidth over ten seconds. However, such long buffers are not possible for low-latency live video streaming and video conferencing. With low/no video buffer, the selected video rate has to closely track the instantaneous network bandwidth to avoid video freeze. To deliver high user QoE for such applications, the available bandwidth has to be accurately estimated in realtime to guide video rate adaption. Over-estimate will lead to video freeze, and under-estimate will lead to unnecessarily low video quality. Various realtime bandwidth prediction (for the next second) algorithms have been adopted by video streaming systems to guide video rate adaption [28, 31, 11, 32]. The emerging VR/AR/MR applications also hinge on high-rate and low-latency delivery of 360° video/virtual objects over mobile connections to facilitate seamless integration of physical and virtual worlds and support user interactions. Realtime bandwidth prediction will again play an important role there. Another direction to cope with bandwidth variations is *multipath transmission*, such as MPTCP [23]. Realtime bandwidth prediction can be used by multipath routing algorithms to *proactively* adjust the traffic split ratios among all the available paths. In this paper, we study realtime prediction of available bandwidth and handoff in 4G/5G mobile networks. *Our first effort is to develop deep-learning based realtime bandwidth prediction models that generate predictions for the available bandwidth in the next few seconds based on the past bandwidth measurement as well as wireless channel and context information.* Specially, we focus on *fixed-route mobility* scenarios, covering routine daily commutes through public transportation and self-driving. We collect long consecutive 4G/LTE traces with a rich set of features in New

lifan@nyu.edu (L. Mei); yongliu@nyu.edu (Y. Liu)
lifan@nyu.edu (L. Mei)
ORCID(s): 0000-0003-1188-481X (L. Mei)

York City MTA public transportation system. Through feature analysis, we identify features with significance for predicting future bandwidth. We then demonstrate that Long Short Term Memory (LSTM) Recurrent Neural Networks [9], in particular TPA-LSTM [25], can effectively mine the latent temporal patterns embedded in channel and context information for accurate *multivariate prediction*. Our LSTM-based prediction models consistently outperform the conventional univariate and multivariate prediction models in both 4G and 5G bandwidth traces.

For handoff prediction, due to the limited initial 5G coverage, and the different deployment plan for 5G and 4G/LTE on urban and rural areas, 5G and 4G/LTE will co-exist in the long time. During an application session, a mobile device will likely switch back-forth between 5G and 4G access modes. Due to the vast disparity between QoS offered by 4G and 5G, it is of even greater importance to predict handoffs between the two access modes in realtime so that applications can make adjustments in advance to anticipate dramatic QoS changes resulted from handoffs. As the example of self-driving vehicle, 5G/4G brings different latency, which triggers totally disparate control strategies. *Our second effort is to predict handoffs between 4G and 5G access modes based on realtime measurement of bandwidth and channel/context information*. We propose two versions of handoff prediction: in *binary prediction*, we predict whether the device will handoff from 4G to 5G or vice versa in the next second; in *continuous prediction*, we predict the probability/fraction of 5G access in a short future time window. We demonstrate that Gradient Boosting Machine (GBM) [4] based classification and regression can achieve high accuracies in binary and continuous 4G/5G handoff predictions.

The rest of the paper is organized as the following. The related work on realtime bandwidth prediction is reviewed in Section 2. We motivate and define the realtime bandwidth prediction problem for fixed-route mobility and present our LTE dataset in Section 3. TPA-LSTM prediction model is introduced in Section 4, followed by prediction accuracy comparison with baselines. In Section 5, we first present 5G bandwidth prediction results, then introduce the handoff prediction problem and present GBM-based classification and regression models for binary and continuous handoff prediction. The paper is concluded with future work in Section 6.

2. Related Work

Realtime bandwidth prediction has been a challenging problem for the networking community.

Authors of [11] and [32] used the Harmonic Mean of TCP throughput for downloading the previous chunks as the TCP downloading throughput prediction for the next chunk. A simple history-based TCP throughput estimation algorithm was proposed in [8]. Authors of [14] used an adaptive filter, Recursive Least Squared (RLS), to make realtime bandwidth prediction for the cellular scenario. For the conventional statistical and machine learning models, in [19], the authors proposed that training a Support Vector Regression (SVR) model [26] to estimate TCP throughput. In the context of DASH video streaming, in [28], authors adopted prediction algorithm in [8] to guide the realtime chunk rate selection, and used a customized SVR model similar to [19] for DASH server selection. In the context of video conferencing, in [31], the cellular link is modeled as a single-server queue driven by a doubly-stochastic service process, and future bandwidth prediction is generated by probabilistic inference based on the single-server queue model. Authors of [27] used Hidden Markov Model (HMM) for bandwidth prediction. Authors of [33] proposed a Random Forest framework to make realtime LTE bandwidth prediction based on the context information. The conventional statistical or machine learning methods are based on short sequence history, and it is not easy to dig out the temporal patterns embedded in rich and complex information structures. For deep leaning methods, in [24], a Deep Neural Network (DNN) based method is applied for bandwidth burst prediction for Human to Machine (H2M) communication. [30], [17], and [15] developed a Long Short Term Memory (LSTM) [6] based method to estimate future bandwidth based on past bandwidth measurements. In [18], authors discussed the feasibility of LSTM models on generalized application scenarios. For the multivariate time series we are facing, complex and non-linear inter-dependencies between variables at different time steps complicate the prediction task. Instead of the vanilla LSTM, we adopt the recently proposed Temporal Pattern Attention LSTM (TPA-LSTM) [25] for bandwidth prediction. It applies attention mechanisms to select the most relevant time steps and variables for prediction. For handoffs, [5] and [12] studied handoff prediction between cells within the same access mode (LTE or 3G). We study handoffs between different access modes, specifically, between 4G and 5G.

3. Realtime Bandwidth Prediction under Fixed-route Mobility

3.1. Fixed-route Mobility

Most people's daily network access patterns are rather predictable. He/She is either at home or office or on the way between the two, as shown in Figure 1. At home and office, there is usually good WiFi coverage, leading to good network Quality of Service (QoS). For the commute between home and office, either driving or taking public transportation, the routes are also relatively fixed. In Metropolitan areas, commuters access the Internet through mobile 4G/5G connections. The mobile access bandwidth is inherently time-varying, especially with user mobility. It is, therefore, important to predict future bandwidth to deliver a high-level of application QoE to commuting users. We choose to focus on realtime bandwidth prediction for fixed-route mobility not only because it covers a wide range of daily commute scenarios, but also it enables a DNN model to learn the bandwidth variation regularity resulted from fixed-routes for accurate prediction.

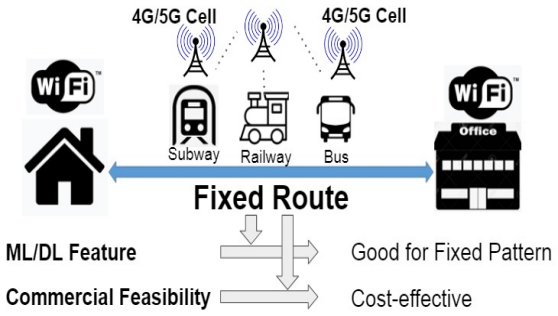


Figure 1: BW Prediction for Fixed-route Mobility

3.2. Realtime Bandwidth Prediction

Let $b(t)$ be the bandwidth available for a user at time t. User mobile device periodically measures the quality of the mobile access link with a certain frequency, e.g., every 1 second. It can obtain a discrete time series of $\{X(t), t = 1, 2, \cdots.\}$, where $X(t) \in \mathbb{R}^n$ is a vector of n types of measured information, including $b(t)$ and other metrics about the connection. The realtime bandwidth prediction problem at time t is to estimate the available bandwidth at some future time instant $t+\tau$ given all the collected measurements up to t:

$$\hat{b}(t+\tau) = \mathbf{f}\left(\{X(k), k = 1, 2, \cdots, t\}\right), \tag{1}$$

where τ is the future horizon value. For prediction based on recent history, we use only $\{X(t-w+1), \cdots, X(t)\}$ to predict $b(t+\tau)$, where w is the sliding window size. In *univariate bandwidth prediction*, we only use past bandwidth measurement to predict future bandwidth, namely

$$\hat{b}(t+\tau) = \mathbf{f}^{(\mathbf{u})}\left(\{b(k), k = 1, 2, \cdots, t\}\right). \tag{2}$$

In a *multi-variate bandwidth prediction*, we use measured channel and context data in addition to the past bandwidth measurement for the prediction.

For univariate bandwidth prediction, there are many methods to construct the prediction function $\mathbf{f}^{(\mathbf{u})}(\cdot)$, from simple history-repeat, i.e., $\hat{b}(t+\tau) = b(t)$, Exponential Weighted Moving Average (EWMA), $\hat{b}(t+1) = (1-\alpha)\hat{b}(t) + \alpha b(t)$, Harmonic Mean, $\hat{b}(t+\tau) = h/\sum_{k=0}^{h-1} 1/b(t-k)$, etc., to more sophisticated signal processing approaches, such as Kalman filter [2] and Recursive Least Squares (RLS) [7]. In [14], RLS is used for realtime bandwidth prediction. By assuming $\hat{b}(t+1) = \sum_{k=0}^{h-1} \omega(k) b(t-k)$, RLS can recursively find the coefficients ω that minimize the weighted linear least squares cost function. [14] showed that RLS achieves higher accuracy than other averaging and signal processing algorithms, such as Least Mean Square and EWMA etc. For multivariate bandwidth prediction, machine learning tools, such as Support Vector Machine [19] and Random Forest [33], have been proposed. In [17] and [30], they show that LSTM deep neural networks can achieve higher accuracy than the conventional bandwidth prediction methods.

3.3. High-level Design and Rationale

For fixed-mobility, we propose a Deep-learning based smart agent for mobile bandwidth prediction. Our framework consists of *data measurement*, *model training* and *model running* steps. Specifically, in the measurement step, for a fixed commute route, the agent repeatedly takes measurements on multiple trips from the start to the end. Bandwidth and related metrics are recorded. In the model training step, one DNN model is trained offline for each commute route, using all the data collected from that route. For the running step, the agent picks the model trained for the current route to generate realtime bandwidth prediction at each time step based on recent measurement.

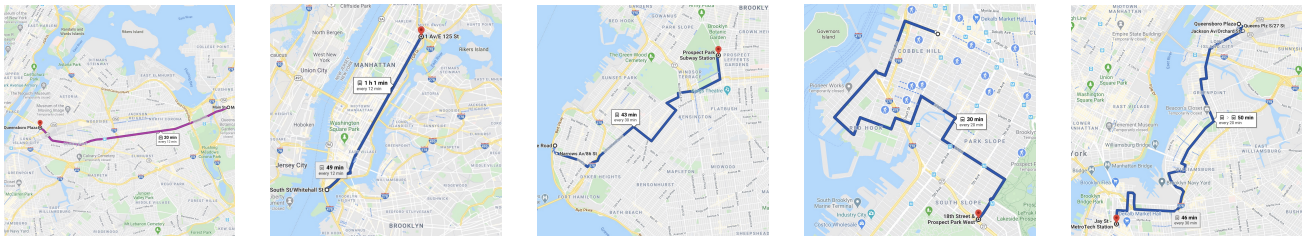
Deep Learning for Prediction: Deep Neural Networks (DNNs) have recently gained lots of momentum due to the dramatic increase in data volume and computing power. They have become the new state-of-art in specific fields, such as computer vision, speech recognition, and natural language processing, etc. What they have in common is *strong temporal and spatial patterns*.

In particular, LSTM-based deep learning method has an unparalleled advantage over the conventional time series analysis tools due to its special recurrent kernel structure. *We explore the temporal patten of mobile bandwidth variations over fixed-routes using LSTM-based DNN.*

Easy Adoption: Users of public transportation systems have strong needs for realtime bandwidth prediction. For a user watching online video, the video player can adapt the quality of the video to be downloaded based on the realtime downlink bandwidth prediction. For a user in a video conference, the conferencing app can dynamically change the resolution and frame rate of the video to be coded and uploaded to other users in the conference, based on realtime uplink bandwidth prediction. Under our proposed smart agent, using off-the-shelf software and hardware, any ordinary smartphone can easily take the measurements. In practice, measurement can be done by network optimization team, by crowdsourcing, or even by transportation company. The offline trained prediction models can run in realtime on local phone or on edge to improve user QoE in various mobile apps.

3.4. LTE Dataset in NYC

Towards our goals, we conducted a measurement study on the public transportation system of NYC. Different from other LTE mobile bandwidth datasets, our dataset is multivariate and focused on fixed public transportation routes. We pick five bus/subway routes of the NYC MTA system, as illustrated in Figure 2a to 2e. The LTE data is collected from Nov, 2019 to the end of Jan, 2020. For each route, we collected long uninterrupted traces by taking around eight trips from one end to the other in both directions, with the duration of each trip to be more than 30 minutes. The total duration of our LTE dataset is around 30 hours. We measured bandwidth, channel, and context related information using $NetMonitorPro$, a mobile network monitoring tool designed for Android devices. We installed this app on a Google Pixel 1 phone with unlimited 4G LTE Data Plan. To

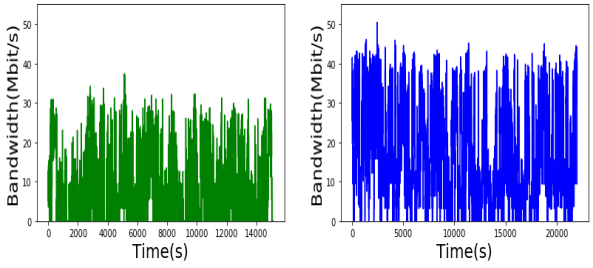


(a) Subway 7 Train (b) Bus M15 (c) Bus B16 (d) Bus B61 (e) Bus B62

Figure 2: Sample LTE Measurement Routes of Public Transportation System of New York City

Table 1
Statistics of NYC Public Transportation Bandwidth Traces (Mbps)

	7 Train	Bus B16	Bus B61	Bus B62	Bus M15
Average	8.67	13.71	17.80	18.34	20.63
Median	6.85	12.80	16.00	15.80	19.10
Max	37.40	45.00	47.40	50.40	44.30
Min	0	0	0	0	0
Std	8.29	9.55	11.69	12.16	9.53
Length (s)	15,116	22,277	21,174	22,000	23,103



(a) Subway 7 Train (b) Bus B62

Figure 3: Sample LTE Bandwidth Traces from Subway #7 and Bus #B62

Table 2
Selected Features

Feature	Information Captured
`Bandwidth(BW)`	download throughput in Mbps
`LTE-neighbors`	number of LTE cells the device can switch to
`RSSI`	power level of received signal
`RSRQ`	quality of received signal
`Echng(Ech)`	whether ENodeB has changed from previous second
`TA`	time advance needed to reach the ENodeB
`Speed`	moving speed of device
`Band`	frequency band of signal

record the bandwidth, we run $iPerf$ to download data from our lab server located on our NYC campus, and record TCP download throughput every 1,000 millisecond. The bandwidth is logged on the mobile phone with timestamps, so we can use timestamps to match the channel and context information logged by $NetMonitorPro$. Figure 3a and 3b visualize two collected bandwidth traces, one from a subway, one from a bus. Table 1 presents the bandwidth statistics of the collected traces.

3.5. LTE Feature Analysis

Other than the bandwidth, we recorded a total of 52 mobile channel and context related features. Out of all the features, we conduct feature selection by calculating the cross-correlation between each feature and bandwidth. Only eight features are selected as the input for the prediction models, as illustrated in Table 2. We further analyzed the importance of each selected feature for bandwidth prediction using Random Forest [16]. We use all the eight features at $t-1$ as input of a Random Forest model, the output is the bandwidth at t.

Table 3 shows the importance weights of all features on all traces.

As expected, `Bandwidth` at $b(t-1)$ has the highest prediction weight on $b(t)$. `Band` has the second highest weight. This is because high speed LTE data transmission is provided in frequency bands like Band 1900 and 2100, while relatively low speed transmission is provided in others, like Band 700. Bandwidth tends to be high when the signal is transmitted in an ideal band, and becomes low when switched to a non-ideal band. The third most important feature is `RSSI`, which indicates whether the signal power between the base station and the mobile device is strong or not. Meanwhile, feature `RSRQ`'s weight is not as high, since it is calculated from `RSSI`. The importance weights of the remaining features are not very high. But this does not mean they are not important for bandwidth prediction. It is only that when all eight features are used together, their prediction power is dominated

Table 3
Feature Importance on LTE Dataset

	Bandwidth	Band	RSSI	RSRQ	TA	LTE_neighbors	Speed	Echng
7 Train	0.5805	0.1105	0.1189	0.0601	0.0393	0.0540	0.0338	0.0028
Bus B16	0.6698	0.1382	0.0612	0.0283	0.0585	0.0195	0.0230	0.0014
Bus B61	0.5170	0.1253	0.1890	0.0634	0.0660	0.0192	0.0184	0.0017
Bus B62	0.5392	0.2310	0.1212	0.0294	0.0491	0.0119	0.0173	0.0009
Bus M15	0.6062	0.2383	0.0497	0.0281	0.0178	0.0269	0.0319	0.0010

by other more powerful features, such as Bandwidth, Band, and RSSI. But they still might provide complementary information in certain scenarios. For example, Echng indicates handoff of ENodeB. Since our device keeps moving, the handoffs occur frequently. Due to handoff, there is a short period of time when the device receives no service from any ENodeB and bandwidth dips to zero. We recorded the ID of the ENodeB that our device is connected to. Whenever the ENodeB ID changes, we consider a handoff happens. Similarly, Speed should be an important feature for mobile bandwidth prediction. Both the signal quality and handoff frequency are highly dependent on device moving speed. But its importance weight calculated by Random Forest is not high. This suggests that its impact on bandwidth is indirectly carried by BAND/RSSI and Echng. *We feed all the eight selected features to our DNN models, which are expected to exploit the complementary prediction power of all features (better than Random Forest) for more accurate prediction.*

4. TPA-LSTM based Bandwidth Prediction

4.1. Introduction to TPA-LSTM

We pick Temporal Pattern Attention Long Short-Term Memory (TPA-LSTM) [25] as our core DNN prediction model. TPA-LSTM extends the vanilla LSTM [9] with the Attention Mechanism [29]. Figure 4a shows the internal structure of the vanilla LSTM unit. LSTM shows great performance in time series analysis due to its special internal memory cell, forget gate, input gate, and output gate. Due to recurrent update, LSTM can keep “long memory" of a time series. Through training, LSTM can learn which part of a time series is important and which is not for predicting the output. Meanwhile, the conventional Attention Mechanism looks back information from the previous time steps, uses the relevance to improve the prediction accuracy. But it is difficult to deal with long time sequences. TPA-LSTM [25] combines the merits of LSTM and Attention Mechanism. It uses a set of Convolution Neural Network (CNN) filters to extract time-invariant temporal patterns. It makes the dimension of attention to be feature-wise, so that it can learn the inter-dependencies among multiple features over a long history time window. Figure 4b illustrates the architecture of TPA-LSTM. Let h_t be the LSTM hidden state at time t. Instead of using h_t to generate prediction for $t+1$, TPA-LSTM learns the “importance" of the hidden states $\{h_{t-1}, \cdots, h_{t-w}\}$ of the previous w steps. Specifically, k 1-D CNN filters with length w are applied to $\{h_{t-1}, \cdots, h_{t-w}\}$, as shown in Figure 4b. Each CNN filter makes convolution over hidden feature values. All filters produce a new matrix H^C. The attention part calculates the attention (importance) weights of all the convoluted hidden features. The scoring function calculates a weight for each row of H^C by comparing it with the current hidden state h_t. The rows of H^C is then weighted summed by their corresponding weights to get a new vector V_t, which is concatenated with h_t to generate an updated hidden state h'_t for the final prediction [25]. The CNN filters and attention calculation enhance the capability of vanilla LSTM to mine periodic temporal patterns in time series. Through experiments on Multivariate Time Series datasets, such as currency exchange rate among several countries and electricity among multiple clients, it has been demonstrated that TPA-LSTM can achieve higher accuracy than LSTM in multivariate time series prediction, even when the periodic pattern is weak [25]. For our mobile bandwidth prediction problem, in some cases, the future bandwidth may be more dependent on history data farther back than the most recent history. TPA-LSTM, using its CNN, can look back further to dig out the inter-dependencies among multiple variables that are multiple time steps apart. In addition, TPA-LSTM perfectly suits the fixed-route mobility scenarios under study. As mentioned in Section 3.3, *for repeated long trips along the same route, periodic patterns on bandwidth, and the related channel features are expected within a single trip and cross multiple trips. TPA-LSTM can effectively mine those patterns for more accurate prediction.*

4.2. Prediction Performance

4.2.1. Methods for Comparison

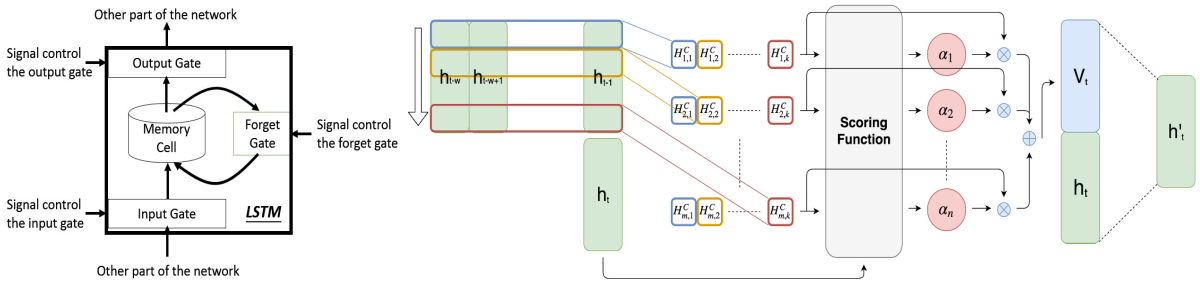
Here we make a comparison between TPA-LSTM method and other baselines univariate and multivariate methods:

- RLS: Recursive Least Square adaptive algorithm [14]
- RF: Random Forest [33]
- LSTM: Vanilla Long Short-Term Memory [30]

Among them, RLS is for univariate with previous bandwidth measurement as its only input feature. The rest of the methods are for multivariate bandwidth prediction, with all the eight features as the input.

4.2.2. Model Settings

For neural network methods, the network structure is: 3 layers, and every layer has 32 units with dropout 0.2. We divide our LTE dataset into a training set, validation/development



(a) LSTM Unit (b) TPA-LSTM

Figure 4: DNN Architecture: a) Internal Structure of LSTM Unit; b) TPA-LSTM Network

Table 4
Evaluation on Bus M15

M15	Mean:	22.7147	Std:	9.2526
Horizon = 1	RLS	RF	LSTM	TPA-LSTM
RMSE	4.7040	4.4912	4.1899	**4.0038**
MAE	3.3713	3.3682	3.1052	**2.9043**
CORR	0.8647	0.8804	0.8939	**0.9025**
Horizon = 2	RLS	RF	LSTM	TPA-LSTM
RMSE	5.1766	4.7689	4.6514	**4.6102**
MAE	3.6173	3.4236	3.2973	**3.2362**
CORR	0.8357	0.8601	0.8663	**0.8671**
Horizon = 3	RLS	RF	LSTM	TPA-LSTM
RMSE	5.5763	5.2344	5.1524	**5.0779**
MAE	3.8503	3.7594	3.6317	**3.5488**
CORR	0.8087	0.8288	0.8309	**0.8362**

Table 5
Evaluation on Subway Train 7

Train7	Mean:	8.3430	Std:	8.0403
Horizon = 1	RLS	RF	LSTM	TPA-LSTM
RMSE	4.8929	4.6958	4.3964	**4.3483**
MAE	3.4632	3.5091	3.2936	**3.1941**
CORR	0.8003	0.8120	0.8405	**0.8414**
Horizon = 2	RLS	RF	LSTM	TPA-LSTM
RMSE	5.0092	4.8089	4.6065	**4.5884**
MAE	3.4660	3.5488	3.3622	**3.2710**
CORR	0.7901	0.8025	0.8216	**0.8222**
Horizon = 3	RLS	RF	LSTM	TPA-LSTM
RMSE	5.3253	4.9537	4.8869	**4.8781**
MAE	3.7218	3.6556	3.5845	**3.4617**
CORR	0.7610	0.7900	0.7953	**0.7963**

set, and test set, according to ratios of 60% : 10% : 30%. We use Adam [13] as optimizer with the default learning rate of 0.001. The loss function is Mean Square Error (MSE) of the predicted bandwidth. For features unique to TPA-LSTM method [25], we set the CNN filter number to 32. For Random Forest, we use the same model setting as [33]. We set the criterion as Mean Square Error (MSE). The `max-features` is set to be Square Root (SQRT). To obtain good performance, the number of decision trees is set to be $1,200$. The max-depth of the trees is set to be 20. The minimum number of samples to be split is 10. The minimum samples kept in one leaf is set to be 2.

4.2.3. Performance Metrics

We use Root Mean Square Error (RMSE) and Mean Absolute Error (MAE) as the main metrics for prediction errors, and use Pearson Correlation (CORR), ranging from -1 to 1, as the reference metric for sequence correlation between the prediction and the ground-truth.

4.3. Results and Analysis

We compare the performance of all the models at different prediction horizons, $\tau = 1, 2, 3$ seconds. For the history window size, we set W to 5 for RLS, LSTM, and TPA-LSTM. For RF, according to the conclusion from [33], a too large W would decrease the accuracy, so we set $W = \tau$.

Table 4 to Table 5 show the detailed evaluation result on two representative dataset traces Bus M15 and Subway Train 7. Table 6 shows the RMSE from B61, B62, and B16 for Horizon equals 1 to 3. The unit over all datasets is Mbps. Among those tables, bold fonts represent the best one for each metric. We can find that TPA-LSTM is the best method on RMSE, MAE, and CORR almost over all the datasets and all the prediction horizons. Taking $\tau = 1$ as an example, for RMSE, TPA-LSTM is on average 11.7% better than the other methods; the improvement over the second-best method is 3.6%. For MAE, TPA-LSTM is on average 15.28% better than other methods, the improvement over the second-best is 5.8%. It shows that TPA-LSTM fits our bandwidth prediction problem well. Also, for the horizon value, as the horizon becomes longer, the prediction performance of each algorithm gets worse. The decreasing trend is also reflected by CORR values. E.g., in Table 5, CORR of TPA-LSTM decreases from 0.8414 to 0.7963 when horizon increases from 1 to 3.

In addition, we found that the relative performance order is RLS<RF<LSTM<TPA-LSTM in general. As a univari-

Table 6
RMSE for B61, B62, and B16 Bus Traces

RMSE	Horizon = 1				Horizon = 2				Horizon = 3			
	RLS	RF	LSTM	TPA	RLS	RF	LSTM	TPA	RLS	RF	LSTM	TPA
B61	4.3403	4.8865	4.2137	**4.0452**	4.7124	4.8845	4.5162	**4.4607**	5.1202	5.2650	4.9645	**4.8987**
B62	4.8097	4.6408	4.4797	**4.2138**	5.3032	5.0046	4.8975	**4.8286**	5.7675	5.4324	5.4020	**5.3250**
B16	3.7693	5.4915	3.6762	**3.5362**	3.9959	4.7346	3.8984	**3.7348**	4.2284	5.0466	4.0465	**3.9697**

ate algorithm, RLS is not as good as the other multivariate algorithms. Even though RLS is already a good adaptive filter for univariate bandwidth prediction [14], it cannot utilize other useful channel and context information, so it hardly can reach the performance of the other multivariate algorithms. Random Forest is multivariate, and utilizes the channel and context information through feature pattern mining. However, for multivariate time series, it is also very important to mine temporal patterns in the long-term time domain. Random Forest does not have enough mining capacity for the long-term temporal patterns. In [33], it was shown that longer history window size not only cannot help prediction, but could decrease the performance. LSTM-based algorithms are designed to mine both long and short-term temporal patterns. This explains why LSTM-based algorithms are better than Random Forest even though they use the same input features. Taking Train7 in Table 5 as an example, at horizon 1, RMSE of RF is 4.6958, however LSTM and TPA-LSTM are 4.3964 and 4.3483, repectively. The gap is large. For MAE, the gap is also clear. MAE of RF is 3.3591, but for LSTM and TPA-LSTM, MAE are 3.2936 and 3.1941 respectively. LSTM and TPA-LSTM have similar architecture: recurrent neural network between input and output, and use various gates to control input and output. That is why the performance gap between LSTM and TPA-LSTM is smaller than the gaps to the others. The small gain of TPA-LSTM comes from its attention mechanism, which allows it to work with longer time windows and a wider range of features at each time step.

5. Bandwidth and Handoff Prediction in 5G Networks

The fifth generation (5G) mobile networks have started to be deployed for commercial use world-wide since 2019. It will become more and more prevalent in the near future. 5G not only promises higher bandwidth throughput but also lower latency than 4G LTE. Figure 5 from [3] visualizes the 10x reduction of the target end-to-end and air latency from 4G to 5G. At the same time, 5G PHY layer operates at higher frequency bands, e.g., millimeter Wave (24-100GHz), which are more vulnerable to higher free space path loss, blockage loss, and penetration loss [1]. This poses a significant challenge to deliver stable 5G mobile access.

5.1. Realtime 5G Bandwidth Prediction

5.1.1. 5G Dataset

We were planning to extend our measurement campaign to commercial 5G deployment in NYC after finishing our

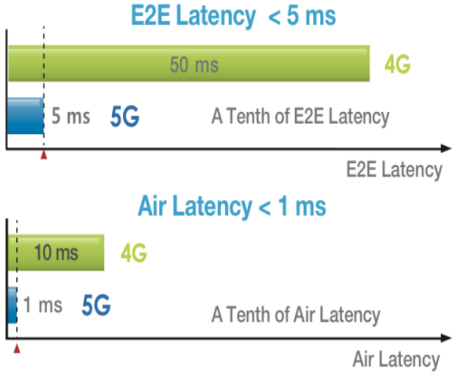


Figure 5: Target Latency in 5G vs 4G [3]

LTE measurement. However, due to the unexpected COVID-19 pandemic, we could not go out for taking measurements in the hardest hit city in the world. We had to turn to 5G datasets collected by other groups before the pandemic. We got to know two recent public datasets; one is from the University of Minnesota (UMN), USA [20], the other one is from the University College Cork (UCC), Ireland [22]. The UMN trace was collected in several metropolitan areas in USA. However, their traces are relatively short, e.g. around 300 seconds per mobility trace, and the channel information and context information were not published. On the other hand, the UCC 5G traces were collected in Cork, Ireland with uplink/downlink bandwidth, as well as rich channel and context information. Even though the traces were not collected on the public transportation system, we picked the car driving traces along fixed-routes (based on analyzing GPS location information) and focus on download bandwidth prediction to make them comparable to our own LTE traces.

Since 5G networks do not have complete coverage yet, the current practice is to fall back to 4G/LTE whenever a UE moves outside of 5G coverage.

In the driving traces, the mobile access mode alternates between 5G and 4G. Table 7 shows the full statistics of bandwidth on the 5G dataset. The first row is the overall statistics for mixed 5G/4G access modes. The total length is 18,043 seconds. The average bandwidth in Mpbs is 39.78. The median is 12.68. The highest is 532.91. The lowest is 0. The standard deviation is as high as 66.73, even twice as the average value! The second row is for 5G access mode only, and the third row is for 4G access mode only. It is obvious that while the 5G access mode has higher average bandwidth than 4G access mode, it also has higher variance than 4G as

Table 7
Statistics of UCC 5G Driving Dataset (Mbps)

	Average	Median	Max	Min	Std	Duration (s)
5G/4G	39.78	12.68	532.91	0	66.73	18,043
5G-only	57.35	19.887	532.91	0	78.95	10,837
4G-only	13.36	8.33	372.53	0	24.77	7,210

Table 8
Selected 5G Features

Feature	Information Captured
DL	downlink throughput in Mbps
UL	uplink throughput in Mbps
RSSI	power level of received signal
RSRQ	quality of received signal
RSRP	reference signal receive power
NRxSRP	RSRP in neighbor cell
NRxSRQ	RSRQ in neighbor cell
SNR	ratio of signal power to the noise power (in db)
CQI	channel quality indicator
NetworkMode	current access mode (5G/LTE)
Cell-handoff	indicator for horizontal handoff between cells of the same mode
Speed	moving speed of device

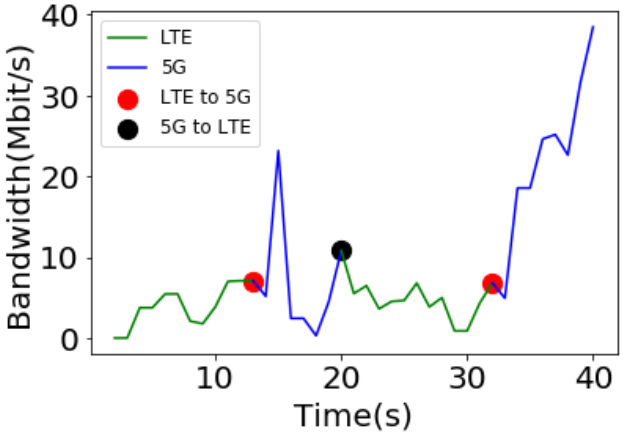


Figure 6: Sample Trace with 5G/LTE Handoffs

expected. It poses a significant challenge for accurate bandwidth prediction, especially when the access mode switches back-forth between 5G and 4G.

5.1.2. 5G Features Analysis

There are totally 26 features recorded in the UCC dataset. Similar to Section 3.5, we selected 12 features for 5G bandwidth prediction using the Random Forest for importance analysis, as shown in Table 8. RSSI and RSRQ have similar meaning as in 4G. NRxSRP and NRxSRQ are signal quality in neighboring cells (NR stands for Neighbor). Cell-handoff is similar to Echng for LTE, indicating handoff between cells. The feature importance for predicting the next-second download bandwidth is reported in Table 9. We can find that the feature with the highest importance is DL, with the importance of 0.585. The second highest is UL with importance of 0.17. The importance of other features are less than 0.10, but they are still taken into account for bandwidth prediction because they have complementary channel and context information for bandwidth prediction.

5.1.3. Prediction Results

Table 10 shows the bandwidth prediction accuracy on the 5G driving trace. The models and parameter settings are the same as in Section 4.2. We can find that on the 5G dataset, TPA-LSTM is still better than the other prediction models. We can also find that the accuracies for all methods are universally worse than their accuracies in our LTE datasets. This is expected because the 5G signal is more dynamic: the mean of the testset is around 27.579, however, the standard variance is as high as 51.276, which is totally different than the five LTE datasets. For Horizon = 3 in Table 10, Random Forest is slightly better than TPA-LSTM in terms of RMSE and CORR, while TPA-LSTM is better in terms of MAE. The large variance of 5G bandwidth and the long prediction horizon of three jointly make the prediction task more challenging, and TPA-LSTM's accuracy improvement is not as big as in the less challenging 4G cases and shorter prediction horizons.

5.2. LTE/5G Handoff Prediction

It is expected that 5G and 4G/LTE will coexist for a long time in the transition phase. A mobile device will frequently switch between 4G and 5G access. Figure 6 shows a sample bandwidth trace of 40 seconds in the UCC 5G driving dataset. The handoffs between 4G and 5G are marked in red circles. The handoffs occur frequently, and the device stays within one mode for only around 10 seconds before switching to the other mode, largely due to the driving speed of 43.65 km/h. Table 11 shows the handoff statistics of the whole driving trace. The average sojourn time with each mode after a handoff is less than 100 seconds and the frequency of handoff is as high as 265 times in a trace of 18,047 seconds.

As witnessed in Table 7, there are apparent bandwidth differences between 5G and 4G accesses. In addition to high bandwidth, 5G is also designed with other new QoS targets, such as ultra-reliable and low-latency communication (URLLC) [10], which is completely absent from LTE networks. 5G is expected to have a QoS leap. In many application scenarios, high data throughput is not the only QoS consideration. For example, in remote surgery and autonomous

Table 9
Feature Importance on 5G Dataset in percentage(%)

DL	UL	NRxSRP	RSRP	NetworkMode	RSSI	NRxSRQ	Speed	SNR	RSRQ	CQI	Cell-handoff
58.50	17.40	5.04	3.95	3.81	2.54	2.41	1.98	1.97	1.21	1.14	0.05

Table 10
Bandwidth Prediction Accuracy on 5G Driving Trace

5G_Driving	Mean:	27.5797	STD:	51.2763
Horizon = 1	RLS	RF	LSTM	TPA-LSTM
RMSE	30.8272	25.2600	25.1736	**24.8120**
MAE	12.6288	10.7901	10.0174	**9.0615**
CORR	0.8016	0.8713	0.8713	**0.8771**
Horizon = 2	RLS	RF	LSTM	TPA-LSTM
RMSE	38.3078	34.5933	34.4389	**33.6579**
MAE	16.1722	16.7128	15.7254	**13.5419**
CORR	0.6734	0.7503	0.7476	**0.7605**
Horizon = 3	RLS	RF	LSTM	TPA-LSTM
RMSE	42.9236	**39.2689**	39.9783	40.2299
MAE	18.4830	20.4666	19.9049	**17.0739**
CORR	0.5680	**0.6547**	0.6506	0.6331

driving, the key is that the command and feedback signals should be sent and received with low latency. In autonomous driving, if the control signals cannot be sent and received in time, it will lead to critical consequences. If one can estimate whether a mobile device will have 5G access in the near future, it will provide valuable information for many applications to adapt their operations. Going back to the autonomous driving example, if one can predict the availability of 5G access, the autonomous driving application can plan ahead for "normal strategy" or "conservative strategy" that works with short or long latencies. For a mobile AR application, the low latency of 5G access can support more frequent user interaction with virtual objects and more real-time feedback, meanwhile the high throughput of 5G access can facilitate data-intensive computation offload to edge servers. We now study 4G/5G handoff prediction.

5.2.1. Handoff Prediction Problem

Let $m(t)$ be an indicator variable representing whether the current access mode is 5G or not. A handoff from 5G to 4G/LTE occurs at t if $m(t-1) = 1$ and $m(t) = 0$. Similarly, a handoff from 4G/LTE to 5G occurs at t if $m(t-1) = 0$ and $m(t) = 1$. To predict handoff at a future time $t+\tau$, we simply need to estimate:

$$\hat{m}(t+\tau) = \mathbf{G}_d\left(\{X(k), k = t-w+1, \cdots, t\}\right), \quad (3)$$

where w is the history window size, $X(k)$ is the past measurements, including $m(k)$. This can be studied as a binary classification problem.

Due to the frequent handoffs back-forth between 5G and 4G, $m(t)$ oscillates between 0 and 1 during the transient period. We introduce a continuous version of the handoff problem. Namely, we introduce a continuous variable $\rho(t)$ as the fraction of time that 5G access is used in a future window $[t, t+\Delta-1]$, i.e., $\rho(t) = \sum_{k=t}^{t+\Delta-1} m(k)/\Delta$. We can then estimate the probability that the device will have 5G access in a future time window starting at $t+\tau$ as:

$$\hat{\rho}(t+\tau) = \mathbf{G}_c\left(\{X(k), k = t-w-1, \cdots, t\}\right). \quad (4)$$

This can be studied as a regression problem.

5.2.2. GBM-based Binary Handoff Prediction

Gradient Boosting Machine (GBM) [4] is an Ensemble Learning method for prediction problems. The main idea of GBM is to ensemble many weak prediction models, generating many sequential decision trees, to build a strong prediction model. To be specific, for inputs that consist of many parameters, every GBM decision tree would generate an output. Outputs of all decision trees are fused to generate the final output. Compared with the other traditional machine learning classification models, Ensemble Learning algorithms are more robust against over-fitting when the number of input features is large. Typically, there are two categories, *Gradient Boosting Classifier (GBC)*, which is to solve classification problems, and *Gradient Boosting Regressor (GBR)*, which is to solve regression problems. In our LTE/5G handoff Prediction, we apply GBC to predict binary handoff events (i.e. whether network mode will change or not), and apply GBR to predict the continuous version of handoffs, (i.e., the chance of 5G access in the near future). We use the Scikit-Learn [21] library to build our GBC and GBR models.

Data Pre-processing: For handoff prediction, we use the same UCC 5G driving dataset [22]. We use the past data from the last 5 seconds, i.e., $w = 5$, to predict whether the access mode will change in the near future. Since handoff prediction is for applications to adjust their operations, to give applications additional time to prepare for upcoming changes, we mark a handoff if the access mode will switch within the next three seconds. Based on that, we create the dataset to train and test our handoff prediction models: extracting all 750 input-output pairs where handoffs happened, and randomly picking 750 input-output pairs where there is no handoff. Among the 750 negative samples without handoff, half of them are temporally close to the positive handoff samples, and the remaining half are far from the handoff samples.

Features for Handoff Prediction:
As discussed, we use data from the last 5 seconds for handoff prediction. For each second, we first consider all the features from our bandwidth prediction experiments. Here, instead of the calculated `Cell-handoff` feature, we use the raw `CellID`. Additionally, we further process the raw `NetworkMode`

Table 11
4G/5G Handoff Statistics in UCC 5G Driving Trace

$5G \rightarrow LTE$	$LTE \rightarrow 5G$	Avg. 5G Sojourn	Avg. LTE Sojourn	STD 5G Sojourn	STD LTE Sojourn
132(times)	133(times)	81.48(s)	54.21(s)	147.25(s)	104.74(s)

Table 12
Different feature sets for our handoff prediction experiments. The BW feature set contains bandwidth features only, and the w.o. BW set contains all features except for the bandwidth features.

Features	# features per second	total features
All	13	65
BW Only	2	10
w.o. BW	11	55

Table 13
Accuracy on Validation Set with Different Learning Rates (All features)

Learning Rate	0.01	**0.04**	0.0475	0.05	0.0525
Average Accuracy	0.737	**0.767**	0.751	0.766	0.763
Learning Rate	0.055	0.1	0.25	0.5	0.75
Average Accuracy	0.756	0.759	0.753	0.740	0.724

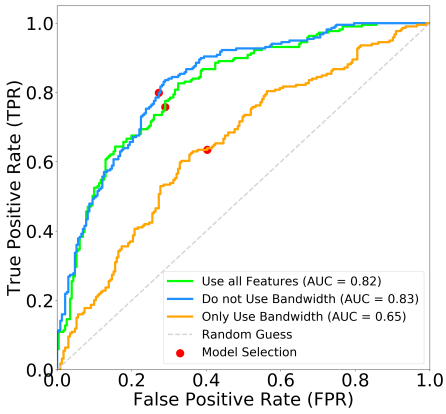


Figure 7: ROC Curve for Unified Model

feature in the previous steps to inform the model whether the access mode has switched in the recent past. Finally, we concatenate the second-wise features from 5 seconds together. This results in 13 x 5 = 65 features in total. Other than using the entire features, we also try two other feature combination sets to predict handoffs: 1) only use the bandwidth features `UL_bitrate` and `DL_bitrate` as input features; 2) use all features except for `UL_bitrate` and `DL_bitrate` bandwidth features. Table 12 summarized the different feature sets for our handoff prediction experiments. We also tried to add statistical features, such as average, variance, median, etc., to the raw features, but did not achieve significant improvement. We just use raw features in the rest of the study.

Parameter Tuning: In Gradient Boosting Classifier, we mainly tune four parameters: `n_estimators`, `learning_rate`, `max_features`, and `max_depth` in the training stage.

For example, for `learning rate`, we tried different values. For each candidate learning rate, we do 5-fold validation, where we will get 5 disjoint train-validation set split. For each split, we compute the prediction accuracy when the model built on the training set is run on the testset. We then compute the average accuracy of these five disjoint configurations as the performance of this specific learning rate. As we can see at Table 13, for experiments that use all features, when the learning rate equals 0.04, we can get the highest average accuracy, which is 0.767. So, we set the learning rate as 0.04. Similarly, the best learning rate for only using bandwidth features is obtained as 0.1, and the best learning rate for the feature set without bandwidth features is 0.0475. Similarly, we set the other parameters as: `n_estimators` = 500, `max_features` = 65 (10 for BW Only feature set, and 55 for w.o BW feature set); and `max_depth` = 8.

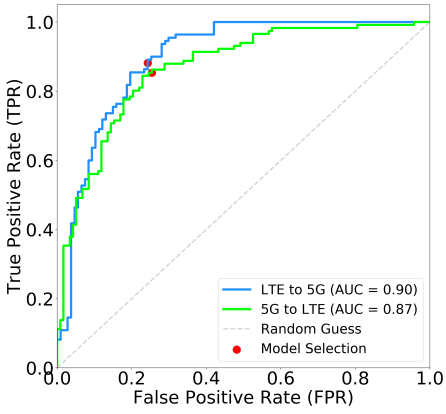


Figure 8: ROC Curve for Separated Models

Table 14
Confusion Matrices, the values within each cell are for different feature combinations: All/BW/w.o. BW

	Predicted 0	Predicted 1
Actual 0	164/138/**168** (TP)	67/93/**63** (FP)
Actual 1	53/80/**44** (FN)	166/139/**175** (TP)

Results: For the dataset, 70% of the data is in the training set, the rest 30% is in the test set. The confusion matrices for three feature sets are compared in Table 14. "0" represents no handoff, and "1" represents handoff. Table 15 reports the True Positive Rate (TPR), False Positive Rate (FPR), Accuracy, F1 Score for each feature set.
We also draw Receiver Operating Characteristics (ROC) curves

Table 15
Prediction Performance of Different Feature Combinations

Features	TPR	FPR	Precision	Recall	Accuracy	F1
All	0.758	0.290	0.712	0.758	0.733	0.735
BW Only	0.635	0.403	0.599	0.635	0.616	0.616
w.o. BW	0.799	0.273	0.735	0.799	0.762	0.766

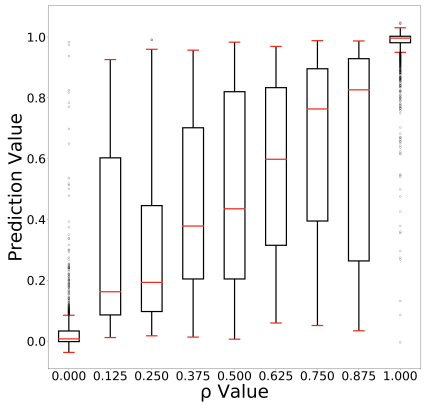


Figure 9: Box Plots for Prediction Accuracy at Different ρ Values

and compute the Area Under the Curve (AUC) for GBC with three feature sets in Figure 7. We also mark the selected TP/FP trade-off point on the curve in red. For the bandwidth-only feature set, the AUC is only 0.65, which means this model has poor discrimination. When we use all features or all except for bandwidth features, the AUC is 0.82 and 0.83, respectively, which means these two models have excellent discrimination. It is clear that only using bandwidth features cannot predict the handoff well. Other channel quality and context features can significantly improve the accuracy. Notice that the green curve and the blue curve are very close. This implies that adding bandwidth information into the feature set does not really improve accuracy (made it slightly worse instead). However, in our previous correlation analysis, bandwidth and mode switch have a high correlation with handoffs. *This suggests that correlation does not necessarily translate into causality. In our setting, handoffs are triggered by channel quality changes, resulted from device mobility, and bandwidth variation is also a consequence of channel quality changes and handoffs. Therefore, it is important to look into channel and context information when predicting handoffs and bandwidth variations in near future.*

5.2.3. Separated 4G → 5G and 5G → 4G Prediction Models

All the experiments above only consider whether network mode will switch, no matter the switch is from 4G to 5G or 5G to 4G. It is expected that the two types of handoffs follow very different patterns. Now we build separated models for two types of handoffs. We divide the handoff datasets into two subsets, one with all samples where the current access mode is 5G, the other one with all samples where the current access mode is 4G. Then we train separated GBC models using 5-fold cross validation and optimal learning rate tuning. The prediction results are shown in Table 16. We also draw Receiver Operating Characteristics (ROC) curve and compute the Area Under the Curve (AUC) for these two models in Figure 8. As expected, while the prediction accuracies for the two types of handoffs are similar, the performance of the separated models are better than the single model for both types of handoffs. *This demonstrates that customized handoff models can better mine the latent characteristics of each type of handoff for more accurate prediction.*

5.2.4. GBR-based Continuous Handoff Prediction

For the continuous version of the handoff prediction, we use all the features from the past 10 seconds as input, i.e., window size $w = 10$, and set the future window size $\Delta = 8$. Then $\rho(t)$ is the fraction of time that the device will have 5G access within $[t, t + 8]$. For example, $\rho = 1$ means the access mode in every second of the next 8 seconds is 5G. We draw boxplots of the predicted $\hat{\rho}$ values for all ground-truth ρ values ranging from 0 to 1, in Figure 9. Most test samples have ρ value of either 0 or 1. As we can see, the prediction errors and their variances for $\rho = 0$ or $\rho = 1$ are small, reflected by the narrow boxes centered around the true value. The variances of predictions for samples with ρ value between 0 and 1 is large, reflected by the wide boxes. In general, the prediction median is in line with the ground truth value ranging from 0 to 1. The RMSE between the predicted value and the ground-truth is 0.109. *This suggests that GBR can predict well the probability/fraction of 5G access in the near future. Such prediction can provide valuable hints for applications to adjust their operations based on the expected QoS metrics in each access mode.*

6. Conclusion

In this paper, we studied the realtime mobile bandwidth and handoff prediction problem using 4G and 5G traces. For fixed-route mobility scenarios, we collected long consecutive traces with rich features, and demonstrated that LSTM, TPA-LSTM in particular, can effectively mine temporal patterns in channel and context information for accurate future bandwidth prediction. For 4G & 5G co-existing networks, we proposed a new 5G/4G handoff prediction problem to mobile networking and multimedia system community. For binary and continuous 5G/4G handoff prediction problems, we developed GBM-based classification and regression models to achieve 80 + % prediction accuracy. As future work,

Table 16
Performance of Separated Handoff Prediction Models

Handoff	TPR	FPR	Precision	Recall	Accuracy	F1
$5G$ to $4G$	0.853	0.254	0.767	0.853	0.799	0.808
$4G$ to $5G$	0.882	0.243	0.789	0.882	0.820	0.833

we will collect a more extensive 5G dataset in NYC MTA system to further improve our prediction models. We also plan to integrate the developed prediction models into low-latency live video streaming and AR application designs to demonstrate how realtime bandwidth and handoff predictions can improve application QoE.

References

[1] Al-Falahy, N., Alani, O.Y., 2019. Millimetre wave frequency band as a candidate spectrum for 5g network architecture: A survey. Physical Communication 32, 120–144.

[2] Brown, R.G., Hwang, P.Y., et al., 1992. Introduction to random signals and applied Kalman filtering. volume 3. Wiley New York.

[3] Burbank, J.L., 2019. 5g vs 4g latency. URL: https://futurenetworks.ieee.org/images/files/pdf/FirstResponder/2019/Jack-Burbank.pdf.

[4] Friedman, J.H., 2001. Greedy function approximation: a gradient boosting machine. Annals of statistics , 1189–1232.

[5] Ge, H., Wen, X., Zheng, W., Lu, Z., Wang, B., 2009. A history-based handover prediction for lte systems, in: 2009 International Symposium on Computer Network and Multimedia Technology, IEEE. pp. 1–4.

[6] Gers, F.A., Schmidhuber, J., Cummins, F., 1999. Learning to forget: Continual prediction with lstm .

[7] Haykin, S., 2008. Adaptive filter theory. pearson education india, in: 27th Annual International Conference of the Engineering in Medicine and Biology Society, IEEE Press. pp. 1212–1215.

[8] He, Q., Dovrolis, C., Ammar, M., 2007. On the predictability of large transfer tcp throughput. Computer Networks 51, 3959–3977.

[9] Hochreiter, S., Schmidhuber, J., 1997. Long short-term memory. Neural computation 9, 1735–1780.

[10] Ji, H., Park, S., Yeo, J., Kim, Y., Lee, J., Shim, B., 2018. Ultra-reliable and low-latency communications in 5g downlink: Physical layer aspects. IEEE Wireless Communications 25, 124–130.

[11] Jiang, J., Sekar, V., Zhang, H., 2014. Improving fairness, efficiency, and stability in http-based adaptive video streaming with festive. IEEE/ACM Transactions on Networking (ToN) 22, 326–340.

[12] Kim, T.H., Yang, Q., Lee, J.H., Park, S.G., Shin, Y.S., 2007. A mobility management technique with simple handover prediction for 3g lte systems, in: 2007 IEEE 66th vehicular technology conference, IEEE. pp. 259–263.

[13] Kingma, D.P., Ba, J., 2014. Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 .

[14] Kurdoglu, E., Liu, Y., Wang, Y., Shi, Y., Gu, C., Lyu, J., 2016. Real-time bandwidth prediction and rate adaptation for video calls over cellular networks, in: Proceedings of the 7th International Conference on Multimedia Systems, ACM. p. 12.

[15] Lee, J., Lee, S., Lee, J., Sathyanarayana, S.D., Lim, H., Lee, J., Zhu, X., Ramakrishnan, S., Grunwald, D., Lee, K., et al., 2020. Perceive: deep learning-based cellular uplink prediction using real-time scheduling patterns, in: Proceedings of the 18th International Conference on Mobile Systems, Applications, and Services, pp. 377–390.

[16] Liaw, A., Wiener, M., et al., 2002. Classification and regression by randomforest. R news 2, 18–22.

[17] Mei, L., Hu, R., Cao, H., Liu, Y., Han, Z., Li, F., Li, J., 2019. Real-time mobile bandwidth prediction using lstm neural network, in: International Conference on Passive and Active Network Measurement, Springer. pp. 34–47.

[18] Mei, L., Hu, R., Cao, H., Liu, Y., Han, Z., Li, F., Li, J., 2020. Realtime mobile bandwidth prediction using lstm neural network and bayesian fusion. Computer Networks 182, 107515.

[19] Mirza, M., Sommers, J., Barford, P., Zhu, X., 2007. A machine learning approach to tcp throughput prediction, in: ACM SIGMETRICS Performance Evaluation Review, ACM. pp. 97–108.

[20] Narayanan, A., Ramadan, E., Carpenter, J., Liu, Q., Liu, Y., Qian, F., Zhang, Z.L., 2020. A first look at commercial 5g performance on smartphones, in: Proceedings of The Web Conference 2020, pp. 894–905.

[21] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al., 2011. Scikit-learn: Machine learning in python. the Journal of machine Learning research 12, 2825–2830.

[22] Raca, D., Leahy, D., Sreenan, C.J., Quinlan, J.J., 2020. Beyond throughput, the next generation: a 5g dataset with channel and context metrics, in: Proceedings of the 11th ACM Multimedia Systems Conference, pp. 303–308.

[23] Raiciu, C., Paasch, C., Barre, S., Ford, A., Honda, M., Duchene, F., Bonaventure, O., Handley, M., 2012. How hard can it be? designing and implementing a deployable multipath tcp, in: NSDI.

[24] Ruan, L., Dias, M.P.I., Wong, E., 2019. Machine learning-based bandwidth prediction for low-latency h2m applications. IEEE Internet of Things Journal 6, 3743–3752.

[25] Shih, S.Y., Sun, F.K., Lee, H.y., 2019. Temporal pattern attention for multivariate time series forecasting. Machine Learning 108, 1421–1441.

[26] Smola, A.J., Schölkopf, B., 2004. A tutorial on support vector regression. Statistics and computing 14, 199–222.

[27] Sun, Y., Yin, X., Jiang, J., Sekar, V., Lin, F., Wang, N., Liu, T., Sinopoli, B., 2016. Cs2p: Improving video bitrate selection and adaptation with data-driven throughput prediction, in: Proceedings of the 2016 ACM SIGCOMM Conference, ACM. pp. 272–285.

[28] Tian, G., Liu, Y., 2012. Towards agile and smooth video adaptation in dynamic http streaming, in: Proceedings of the 8th international conference on Emerging networking experiments and technologies, ACM. pp. 109–120.

[29] Wang, Y., Huang, M., Zhu, X., Zhao, L., 2016. Attention-based lstm for aspect-level sentiment classification, in: Proceedings of the 2016 conference on empirical methods in natural language processing, pp. 606–615.

[30] Wei, B., Kawakami, W., Kanai, K., Katto, J., Wang, S., 2018. Trust: A tcp throughput prediction method in mobile networks, in: 2018 IEEE Global Communications Conference (GLOBECOM), IEEE. pp. 1–6.

[31] Winstein, K., Sivaraman, A., Balakrishnan, H., 2013. Stochastic forecasts achieve high throughput and low delay over cellular networks, in: Presented as part of the 10th {USENIX} Symposium on Networked Systems Design and Implementation ({NSDI} 13), pp. 459–471.

[32] Yin, X., Jindal, A., Sekar, V., Sinopoli, B., 2015. A control-theoretic approach for dynamic adaptive video streaming over http, in: ACM SIGCOMM Computer Communication Review, ACM. pp. 325–338.

[33] Yue, C., Jin, R., Suh, K., Qin, Y., Wang, B., Wei, W., 2018. Linkforecast: Cellular link bandwidth prediction in lte networks. IEEE Transactions on Mobile Computing , 1582–1594.